KLE Technological University
Creating Value
Leveraging Knowledge
BVB Campus, Vidyanagar, Hubballi – 580031, Karnataka, INDIA.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Project report on

Single Image Depth Estimation in Underwater Environment

Submitted

in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

Dinesh C Dhotrad 01FE17BCS069
K Sai Sandeep 01FE17BCS092

Under the guidance of

Dr. Uma Mudenagudi

School of Computer Science and Engineering

KLE Technological University, Hubballi

2020-2021

KLE TECH KLE Technological University

Creating Value Leveraging Knowledge

BVB Campus, Vidyanagar, Hubballi – 580031, Karnataka, INDIA.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

2020-21

CERTIFICATE

This is to certify that project entitled “Single Image Depth Estimation in Underwater Environment” is a bonafied work carried out by the student team Dinesh C Dhotrad 01FE17BCS069, K Sai Sandeep 01FE17BCS092, in partial fulfillment of the completion of 7th semester B. E. course during the year 2020 – 2021. The project report has been approved as it satisfies the academic requirement with respect to the project work prescribed for the above said course.

Guide Name	SoCSE Head
Dr. Uma Mudenagudi	Dr. Meena S. M.

External Viva-Voce

Name of the examiners	Signature with date
1 ______________________	________________
2 ______________________	________________

ABSTRACT

In this project, we address the difficult problem of single image depth estimation for an underwater scenario. The proposed architecture consists of two Generative Adversarial Networks (GANs), the first Network generates an initial estimation of the indefinite structured depth map, whereas the second Network acts as both refinement and contrastive network to generate the final depth map. These networks learn the mapping from the input image to the output image, as well as learn a loss function to train this mapping. For training of our model, we use synthetically generated underwater RGB images and their corresponding depth that have been generated through NYU Depth Dataset V2 for different water types (Jerlov). We demonstrate our results on both the synthetic dataset and real underwater images.

Keywords : *Learning Based Single Image Depth Estimation, Generative Adversarial Network, U-Net*

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of a number of individuals whose professional guidance and encouragement helped me in the successful completion of this report work.

We take this opportunity to thank Dr. Ashok Shettar, Vice-chancellor and to Dr. N H Ayachit, Registrar, KLE Technological University, Hubballi.

We also take this opportunity to thank Dr. Meena S M, Professor and Head of Department, Department of Computer Science and Engineering for having provided us academic environment which nurtured our practical skills contributing to the success of our project.

We sincerely thank our guide Dr. Uma Mudenagudi Dean R&D, Department of Computer Science and Engineering for her guidance, inspiration and wholehearted co-operation during the course of completion.

We sincerely thank our project co-ordinators Dr. P.G Sunita Hiremath and Dr. Sujatha C. for their support, inspiration and wholehearted co-operation during the course of completion.

Our gratitude will not be complete without thanking the Almighty God, our beloved parents, our seniors and our friends who have been a constant source of blessings and aspirations.

Dinesh C Dhotrad - 01FE17BCS069
K Sai Sandeep - 01FE17BCS092

CONTENTS

Chapter 1

INTRODUCTION

Estimating depth from a given single RGB image is a fundamental task in computer vision which is most directly solved using the learning method, the most efficient among them would be supervised deep learning. The main goal in Underwater Monocular Depth Estimation is to predict the depth value of each pixel of a given single RGB image input. Our goal is to generate the most accurate depth map for the given any underwater image. Here we use GANs and cGANs were cGANs are suitable for image-to-image translation tasks such as the prediction of the depth map in our case, where we condition on an underwater input image and generate a corresponding output depth map. The GANs in our model learn a loss that tries to generate and classify the depth map with its accuracy, simultaneously training a generative model to minimize this loss. Blurry edges in the depth map will not be tolerated since they look inaccurate.

1.1 Motivation

Since the need for 3D Reconstruction of Underwater Data is necessary, the generation and gathering of accurate depth a difficult task. The sensors available today don't work well underwater due to the density of water and the sensors calibrated for solving the necessary purpose of above water applications. This requires the need for generating synthetic underwater depth data for the 3D Reconstruction of Underwater Objects captured in Underwater Images. 3D Reconstruction of Poompuhar Underwater Images requires depth estimation. But Underwater Data of Poompuhar doesn't contain nither stereo pair images for the calculation of good Stereo Depth or corresponding depth map for the RGB images so there is a need for robust Learning-Based Depth estimation that helps in the restoration and enhancement of the underwater images in the given dataset.

1.2 Literature Review / Survey

Depth estimation is an important component of understanding geometric relations within a scene. Depth estimation from a single image is described as an ill-posed and inherently ambiguous problem. Recovering depth information in applications like 3D modelling, robotics, autonomous driving is more important when no other information such as stereo images, optical flow or point clouds are available. For the task of depth estimation using single images, learning-based methods have shown very promisingly. Other works suggest using cGANs to map RGB images, as well as sequences of frames and optical flow on ground truth measured using LiDAR [1], using CNN and compare the performance of different CNN architecture on NYU Depth Dataset V2 [2] to generate information for the depth map of the scene.

1.3 Problem Statement

Single Image Depth Estimation by the Supervised Learning Method. With the aim of digital reconstruction and preservation of heritage sites by the Government of India, the collection and preservation of data are challenging due to its mass and diversity. Few heritage sites are buried underwater and left no clues for reconstruction. Capturing depth information underwater is challenging due to the capture conditions and unavailability of sensors that work underwater. Towards this, we plan to model a uwrgb2depth architecture to estimate a depth map from a given single RGB image captured in underwater conditions.

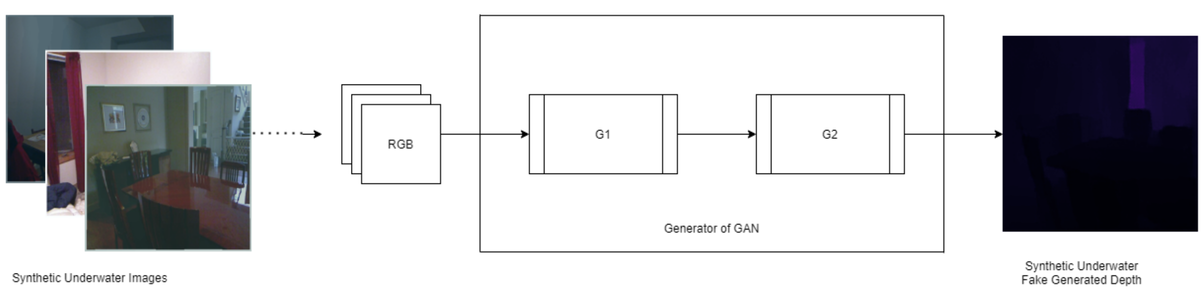

Figure 1.1: Proposed uwrgb2depth architecture

1.4 Problem Analyses

The problem requires the implementation and training of Generative Adversarial Networks which can convert given underwater image to the corresponding depth map. A GAN is implemented using two main components, namely a generator and a discriminator, A generator generates output that is then evaluated by the discriminator were the feedback is sent back to the generator as an input to decrease the loss and improve the generated depth map.

1.4.1 Scops and Constrants

- GANs are heavily dependent computationally powerful hardware.
- The Project focuses on solving real world problem using synthetic
- The dataset used in the project is syntheically generated and the results in the real world may vary.

1.5 Objectives

- Design a Deep Neural Network towards generation of depth map for monocular image captured underwater.
- Demonstration of results on benchmark datasets using different quality metrics and comparision with State of art techniques.

Chapter 2

REQUIREMENT ANALYSIS

For every system, we build there is a need for Requirements Specification which is a set of documentation that describes the features and behaviour of a system or software application. In developing System Requirements, it is important to follow a structured task and activity flow. Such a process ensures the success of the project by providing a clearly defined project scope and a method for managing the scope.

2.1 Functional Requirements

- Alignment of RGB and Depth images: Correction of the discordance of colour information and its corresponding depth in the depth map.
- Indistinct or fake depth images are to be generated by the Generator.
- Discriminator to discriminate between images generated by the generator with ground truth.
- High-quality images should be generated with the help of the patches discrimination approach through PatchGAN.

2.2 Non Functional Requirements

- Computer Hardware
  - Multi-Core Processor, Recommended Intel Core I5, I7.
  - CUDA enabled Nvidia GPU
- Computer Software
  - Anaconda
  - Computer Vision library
  - Tensorflow and PyTorch Framework

2.3 Hardware Requirements

A hardware requirements most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware.Below mentioned Specifications help for our GAN model to train with less amount of duration.

CPU	Intel i5/i7 or equivalent
RAM	64GB
GPU	Nvidia GV 100 or equivalent
Storage	SSD with at least 500GB of storage

Table 2.1: Hardware requirements

2.4 Software Requirements

Below mentioned are the software and APIs required for the train and testing for the proposed GAN architecture.

Recommended OS	Ubuntu or any Linux OS
Library/Framework	Pytorch
Programming language	Python3

Table 2.2: Software requirements

Chapter 3

SYSTEM DESIGN

3.1 Overview

System Design is the process of defining the elements of the system suck as the architecture, modules and components. The different interfaces of those components and the data that goes through the system. The purpose of the System Design process is to provide sufficient detailed data and information about the system and its system elements to enable the implementation consistent with architectural entities as defined in models and views the system architecture. It is the phase that bridges the gap between the problem domain and the existing system in a manageable way It is meant to satisfy the specific needs and requirements of a business or organisation through the engineering of a coherent and well-running system. the system design then overlaps with the systems analyses, systems engineering and system architecture.

3.2 Architectural Framework/System Design

In this section, we discuss our proposed Pix2Pix architecture for the single image depth estimation in the environment as shown in Figure 3.1.now the Synthetic Underwater Images with their Corresponding depth are sent to the General Adversarial Networks(GAN) which has an encoder-decoder type generator with residual transposed convolutional blocks trained with an adversarial loss. Two types of networks are used which are Context Network and Refinement Network. In Content Network Generator generates fakes samples of depth maps and tries to fool the discriminator. In Refinement Network Discriminator tries to distinguish between the real and fake samples. The Generator and the Discriminator are both Neural Networks and they both run in competition with each other in the training phase. The steps are repeated several times and in this, the Generator and Discriminator get better and better in their respective jobs after each repetition.

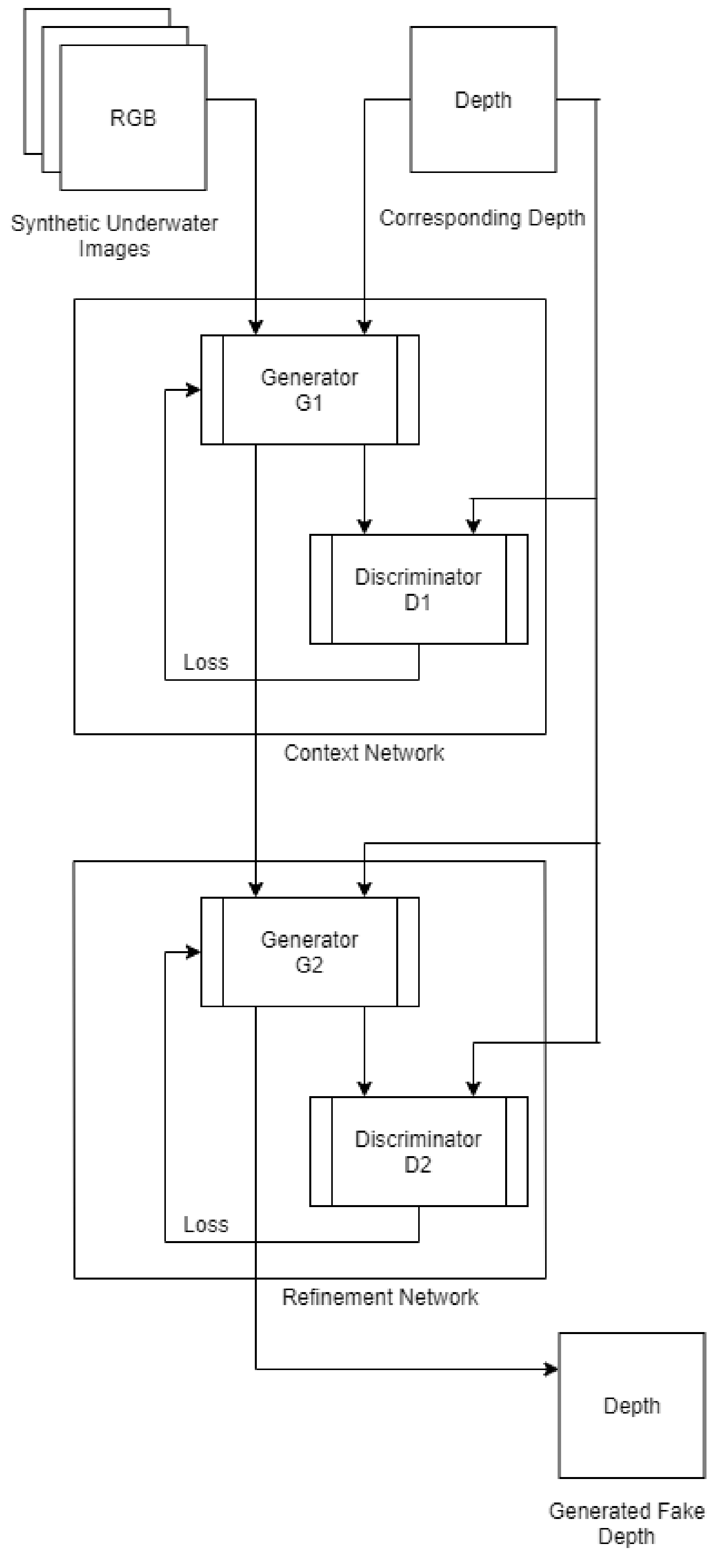

Figure 3.1: High Level Architecture

3.3 Level 0 DFD

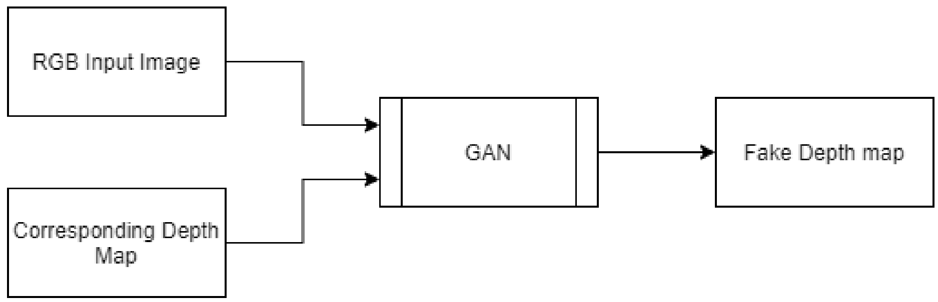

Figure 3.2: Level 0 DFD

Level 0 DFD is a data flow diagram that shows the entire pipeline at an abstract level. Underwater image as a given input. We send it to multiple GAN models to generate a fake depth map with the help of the original depth map of synthetic generated underwater images. Based on the input image the output would be the depth map of it.

3.4 Detalied DFD of the System

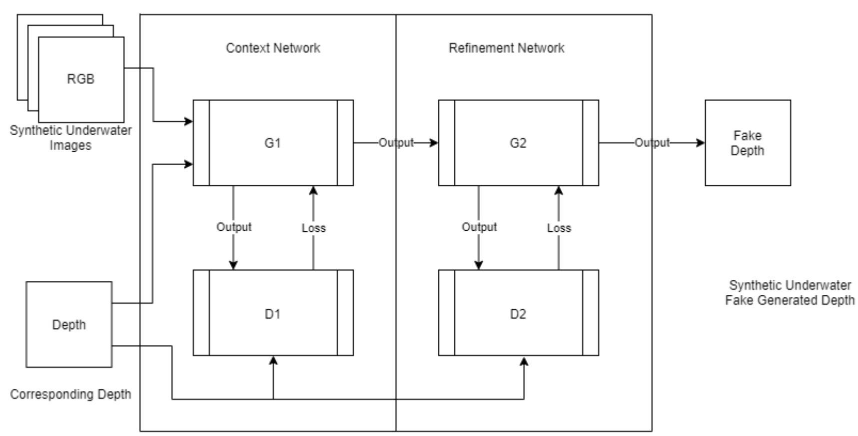

Figure 3.3: Detalied DFD of the System

The Synthetically generated underwater images and the corresponding depth are given as input. The first Generative Advaserial Network generates an initial estimation of the indefinite structured depth map and feeds the output to the second Generative Advaserial Network, this Network acts as both refinement and contrastive model to generate the final depth map which is rendered as output. 8

Chapter 4

IMPLEMENTATION

4.1 Dataset

Our Underwater Synthetic Data-set consists of Synthetically Generated test data of NYU Depth Dataset V2, The NYU-Depth V2 data set is comprised of video sequences from a variety of indoor scenes as recorded by both the RGB and Depth cameras from the Microsoft Kinect. Giving NYU Images with its corresponding depth and taking jerlove tints as B^{∞} in revised image formation module synthetic underwater images are generated.

4.2 GAN Module

Runtime Environment	Nvidia GV100 with Conda Distribution
Framework	PyTorch
Architecture	uwrgb2depth
Modules	Inception Block Pairwise U-Net Skipped Connections Learning based UpSampling(Transpose Convolution with activation)
Activation function	ReLU,LeakyReLU
Batch size	16
Learning rate	0.0001, Weight Decay 1e-5
Number of epochs	40
Loss Functions	BCE, MSE, SSIM
Learning Rate Scheduler	Cosine Annealing
Optimizer	Adam, RMS Props

Table 4.1: Training setup: GAN Module

4.2.1 Generator 1 Module

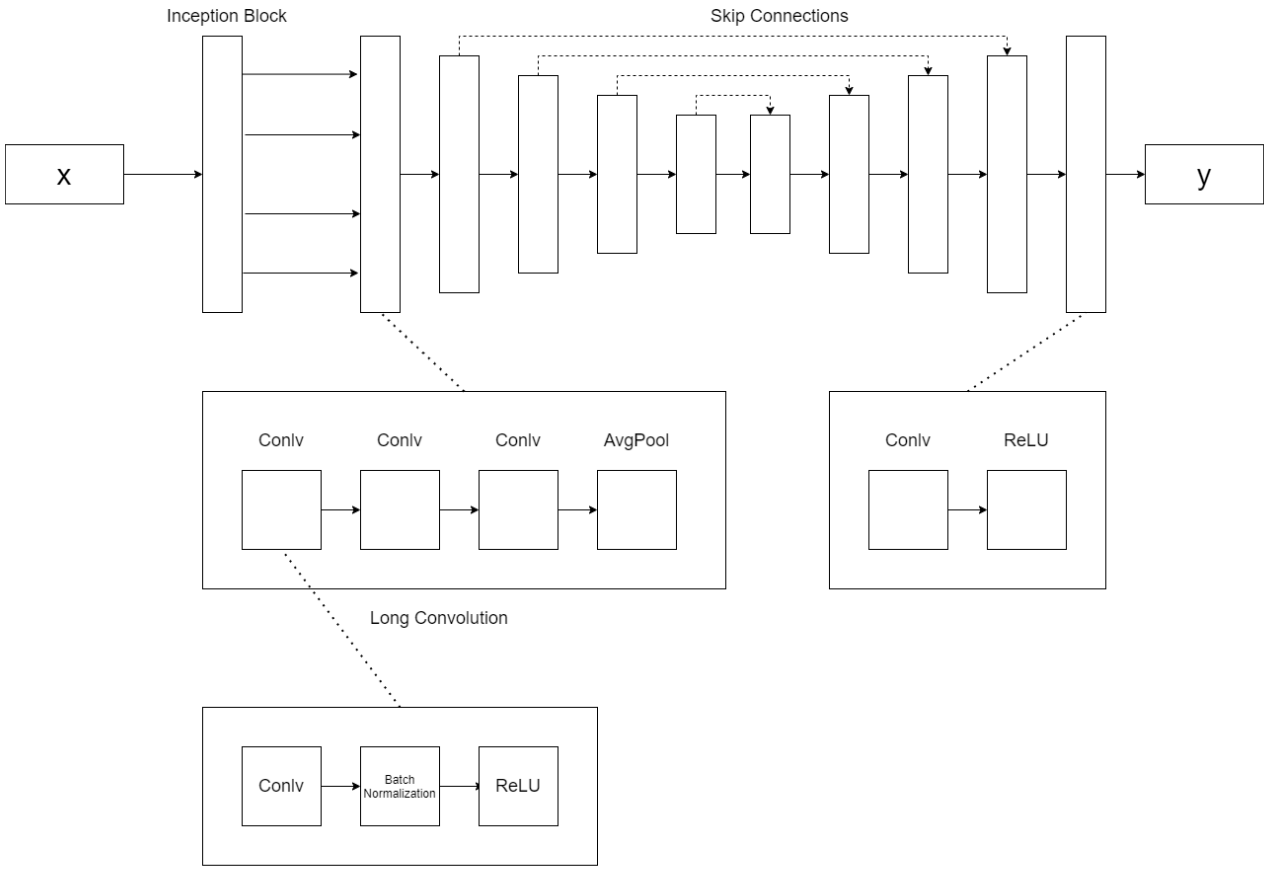

Figure 4.1: Generator 1 Module

The images are sent through the Inception block to extract the features that allow us to use multiple types of filter size, instead of being restricted to a single filter size. After the feature extraction, feature maps are sent to the Encoder-Decoder Architecture where long encoder and short decoder is used in the Generator 1 module. The long encoder consists of three long convolutional blocks and an average pool where each convolutional block consists of a convolutional layer, Batch Norm and Relu function. The features which are extracted from the first Convolutional block, the same amount of features are extracted from the second convolutional block and the third convolutional block, these will generate more features and pass on to the final Average Pool layer. Later the result is passed on to the short decoder which consists of Convolution Transpose and Relu function. The fake depth maps are generated from this Generator architecture are the initial estimation of the indefinite structured depth map and passed on to the Discriminator that generates loss which is later sent back to Generator for backpropagation.

4.2.2 Generator 2 Module

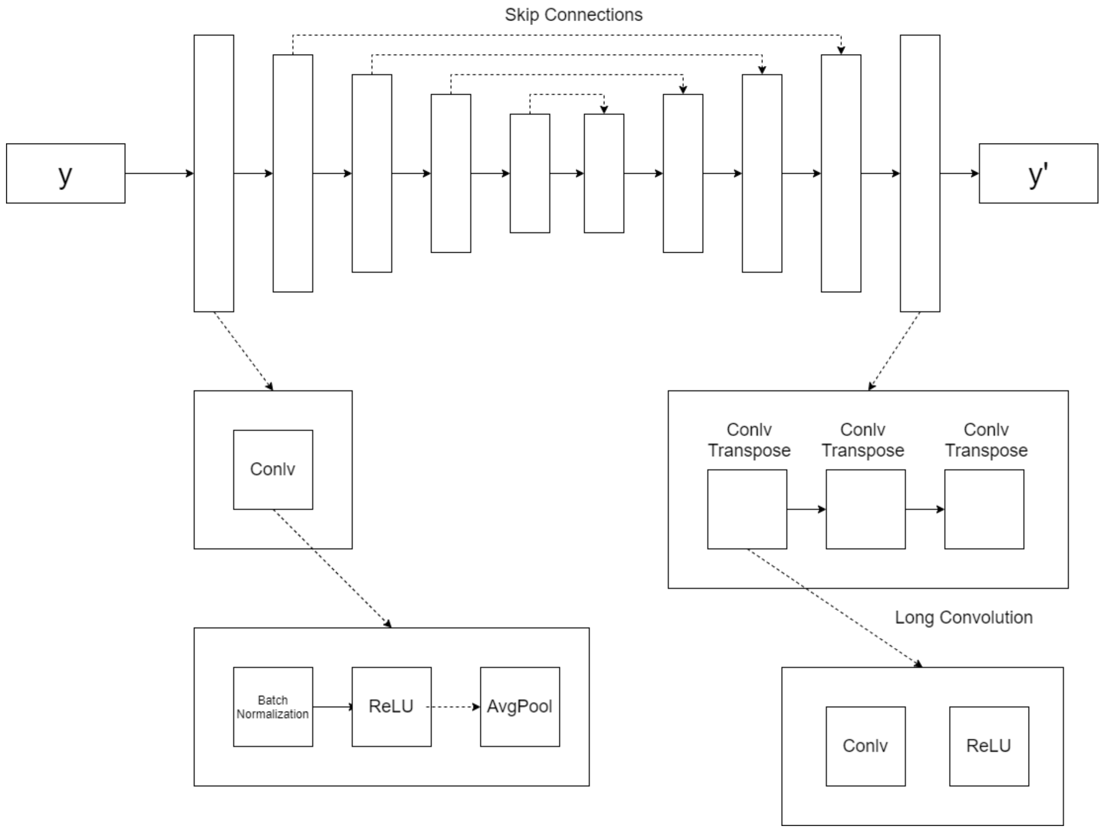

Figure 4.2: Generator 2 Module

This module is the exact reverse of the Generator 1 Module. Here the generator will have a short encoder and long decoder. Result Generated from Generator 1 is later sent to the encoder of Generator 2 which has one convolutional block whereas the decoder will have three Convolutional Transpose with relu functions stating as a short encoder and long decoder. After the fake depth maps get processed through this generator architecture, the Discriminator will try to distinguish between the real and fake depth maps. The discriminator just acts like a classifier that distinguishes the real data from the data generated by the generator. During discriminator training, the discriminator ignores the generator loss and just uses the discriminator loss. The discriminator loss penalizes the discriminator for misclassifying a real instance as fake or a fake instance as real. The discriminator updates its weights through backpropagation from the discriminator loss through the discriminator network.

Chapter 5

RESULTS AND DISCUSSIONS

5.1 Synthetic Underwater Image

Below given are the pictorial representation of generated results consisting of images and depth maps, Figure 5.1, 5.2 and 5.3 each consisting of different epoch stage. The below images contain the Input Synthetic Underwater test Image and corresponding depth image.

Underwater Image | Corresponding Depth | Generated Fake Depth

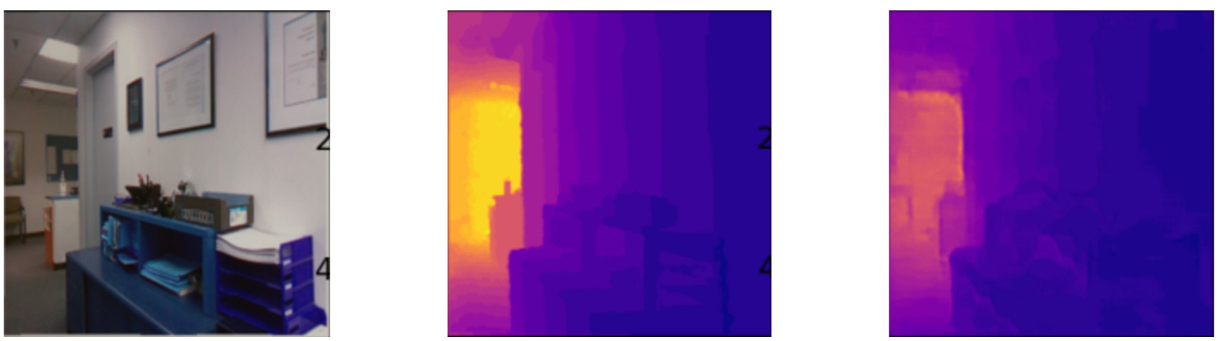

Figure 5.1: Epoch 20

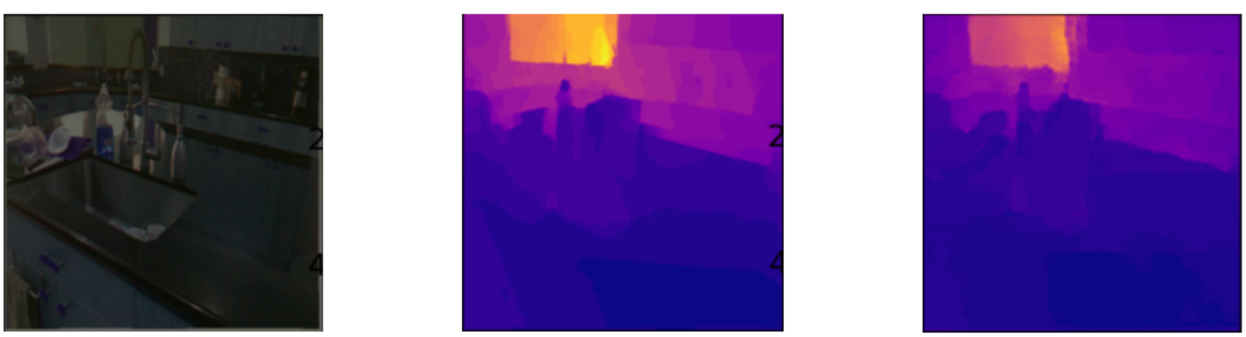

Figure 5.2: Epoch 22

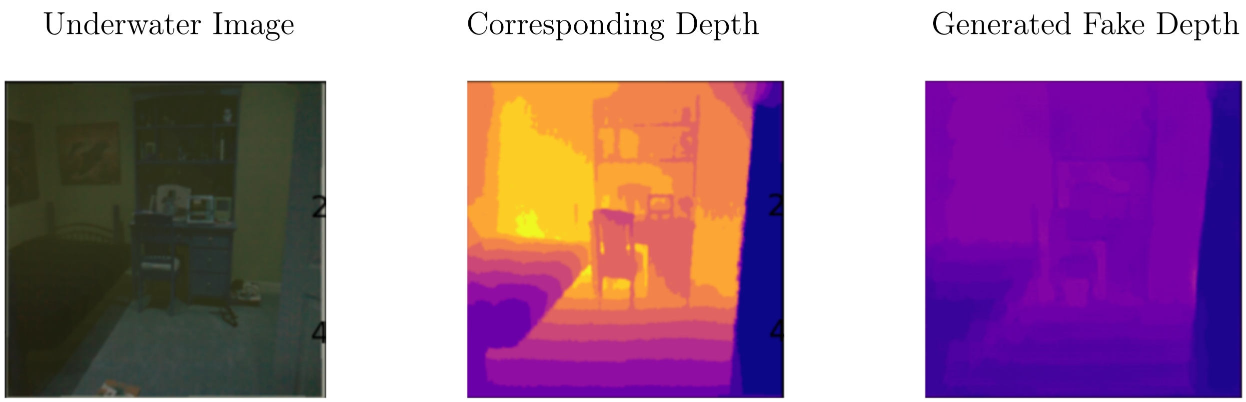

Figure 5.3: epoch 34

The above images show the results of the proposed uwrgb2depth architecture using a Synthetically generated Underwater Dataset on NYU V2 Dataset. In the above-displayed images the images towards the leftmost are input images, towards the centre are actual captured corresponding NYU Depth map and towards the right side is the generated depth map by our architecture.

The above results are compared with a generated depth map and actual depth map and are evaluated using two measuring tools that are widely used in image quality assessment, those are Peak signal-to-noise ratio (PSNR) and Structural Similarity (SSIM). The below table shows the average evaluation result of the three above generated depth maps.

Architecture	Metric	Avg Result
uwrgb2depth	PSNR	48.3433
	SSIM	0.7967

Table 5.1: Comparison Table

5.2 Real Underwater Images

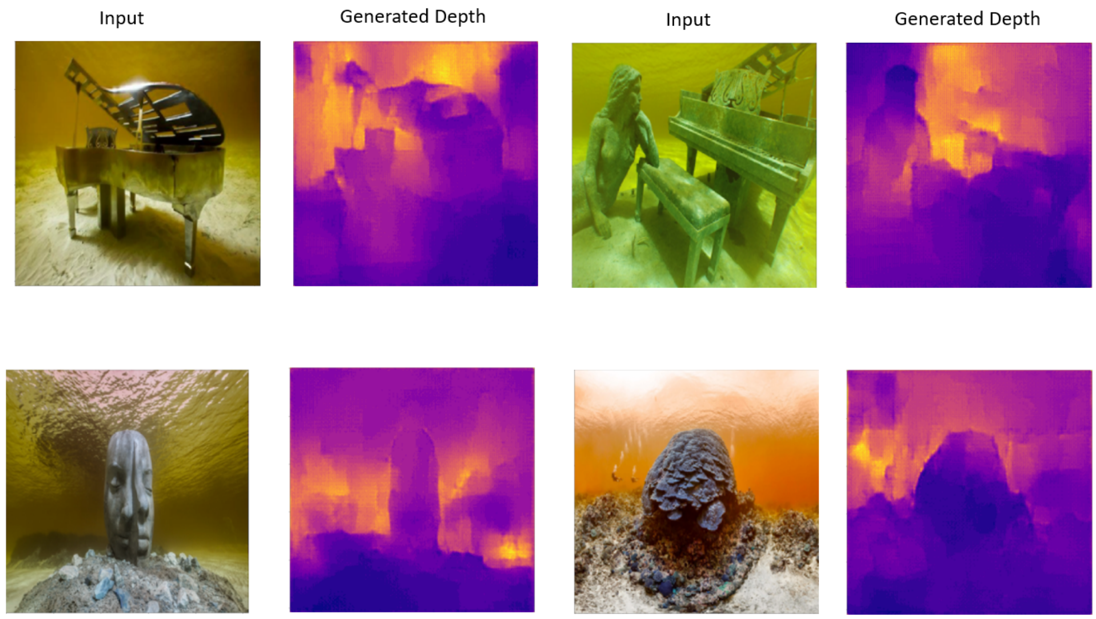

Figure 5.4: Results on Real Underwater Images

The above images show the results of the proposed uwrgb2depth architecture using a real world underwater images resulting in the output of the generated depth map.

Chapter 6

CONCLUSION AND FUTURE SCOPE OF THE WORK

6.1 Conclusion

Depth prediction using a monocular image plays an essential role in many practical applications and is challenging because of the inherent ambiguity. A deep architecture learns the depths of the synthetic images and provides this information to the classifier network. In this project, we have proposed uwrgb2depth architecture using the Conditional Generative Adversarial network on our Synthetic Underwater Image dataset. We have demonstrated the results of the proposed uwrgb2depth architecture using quality metrics. This architecture yields acceptable result while reduces a large number of model parameters.

Further, the module can be applied to train any depth estimation as conditional adversarial networks are a promising approach for many image-to-image translation tasks such as our problem to demonstrate the solution for estimating depth map, especially those involving highly structured graphical outputs which help in 3D scene rendering. These networks learn a loss adapted to the task and data at hand, which makes them applicable in a wide variety of settings.

6.2 Future scope

The above model is complex in architecture resulting in slow learning of the model, these can be solved by reducing the proposed architecture that helps to learn the features of an RGB image faster and outputs depth map result faster. As the model is tuned to learn underwater images and output its corresponding depth map, its suitable for the above water scenario also. Since we rely on multiple GANs in our architecture the reduction and optimisation in our model can be done significantly.

REFERENCES

[1] K. G. Lore, K. Reddy, M. Giering, and E. A. Bernal. Generative adversarial networks for depth map estimation from rgb video. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1258–12588, 2018.

[2] Z. Geng. Depth estimation from single image using cnn-residual network. 2017.

[3] Shir Gur and Lior Wolf. Single image depth estimation trained via depth from defocus cues. 2020.

[4] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. 2018.

[5] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2020.

[6] Simon Niklaus, Long Mai, Jimei Yang, and Feng Liu. 3d ken burns effect from a single image, 2019.

[7] Lijun Wang, Jianming Zhang, Oliver Wang, Zhe Lin, and Huchuan Lu. Sdc-depth: Semantic divide-and-conquer network for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[8] Maxim Maximov, Kevin Galim, and Laura Leal-Taixé. Focus on defocus: bridging the synthetic to real domain gap for depth estimation, 2020.